2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Bayesian meta learning for trustworthy uncertainty quantification

Zhenyuan Yuan[1] Thinh T. Doan[1]

Abstract— **We consider the problem of Bayesian regression with trustworthy uncertainty quantification. We define that the uncertainty quantification is trustworthy if the ground truth can be captured by intervals dependent on the predictive distributions with a pre-specified probability. Furthermore, we propose, Trust-Bayes, a novel optimization framework for Bayesian meta learning which is cognizant of trustworthy uncertainty quantification without explicit assumptions on the prior model/distribution of the functions. We characterize the lower bounds of the probabilities of the ground truth being captured by the specified intervals and analyze the sample complexity with respect to the feasible probability for trustworthy uncertainty quantification. Monte Carlo simulation of a case study using Gaussian process regression is conducted for verification and comparison with the Meta-prior algorithm.**

I. Introduction

Engineering systems operating in the real world are usually subject to unknown uncertainties. Examples include autonomous cars driving in urban scenarios, unmanned aerial robots for outdoor package delivery and mobile robots for search-and-rescue. In order to ensure mission success and the safety of these systems while maintaining high autonomy, it is necessary to learn and quantify these uncertainties in a trustworthy manner such that these systems can operate with minimal human supervision/intervention.

Bayesian learning [1] is a class of statistical learning frameworks, including but not limited to Gaussian process regression (GPR) [2], Kalman filtering [3], Bayesian neural network [4] and particle filtering [5]. In general, Bayesian learning first models a target (e.g., system state, parameter or function) as a sample from a distribution a priori, then given a set of data it utilizes the Bayesian inference framework to compute a posterior distribution of the target for prediction. With proper choice of the prior distribution and mild assumptions on the target, Bayesian learning is able to consistently approximate the target [6] [7]. Established analysis, such as the well-known PAC-Bayesian theorems [8], has shown that the generalization error for the performances of Bayesian learning methods decreases at the rate of $\mathcal{O}(\frac{1}{\sqrt{n}})$, where n is the number of data samples. Furthermore, the predictive distributions, i.e., the prior and the posterior distributions, inherently allow Bayesian models to predict with uncertainty quantification for each input. These aforementioned advantages make Bayesian learning a powerful tool in a variety of applications, e.g., optimization [9], learning-based control [10]–[12], motion planning [13] [14], state estimation [3] and system identification [15].

Based on the Bayes rule [1], the posterior distribution reflects predictive uncertainties accurately only if the likelihood function and the prior distribution are correctly specified, which are also a common assumption in the analysis for uncertainty quantification [16] [17] [18]. However, in some cases, the aforementioned information may not be obtained a priori accurately. To relax this assumption, existing works instead assume the likelihood and/or the prior distribution are specified structurally such that the hyperparameters can be learned through data [2] [19] [20]. While there are powerful function approximation models, such as deep neural networks and finite-dimensional basis functions, which are able to consistently approximate a wide range of targets, in some cases it can be hard to ensure the selected class of approximation models fully capture the target when there is no much prior information about the structure of the target. Furthermore, when the data is scarce, the hyperparameters may not be well learned [21]. As a result, the uncertainty quantification obtained from the predictive distributions may not be trustworthy, and the subsequent operations, e.g., synthesis of safety controller [22] [23] and selection of safe decisions [24] [25], leveraging the uncertainty quantification can be unreliable.

Meta learning is a machine learning framework which aims to utilize the data of a collection of tasks to identify a good initialization/prior for the learning algorithm in new tasks such that fast learning can be achieved using a small amount of data [26]. The process of identifying a good initialization/prior is known as the *meta training* procedure, and the learning in a new task is known as the *adaptation* procedure. The problem is usually formulated as an optimization problem, where the objective function is the expected performance of the adapted model in a new task. Most of the works do not consider uncertainty quantification [26]–[29]. Generalization errors, which can be used to derive for uniform uncertainty quantification, are considered in [30] [31] for adapted models. Bayesian meta learning is considered in [32]–[38], where the posterior distributions provide input-dependent uncertainty quantification. In particular, papers [32] [33] consider meta learning for Bayesian neural networks, where the prior distributions of the parameters in the neural networks are meta trained to optimize the performances of the neural networks with parameters sampled from the posterior distributions. Papers [34]–[38] consider Gaussian process or Bayesian linear regression, where the hyperparameters in the prior covariance and/or mean (functions) are meta trained provided pre-specified structures.

[1]Zhenyuan Yuan and Thinh T. Doan are with Bradley Department of Electrical and Computer Science, Virginia Tech, Blacksburg, VA 24061, USA (email: {zyuan18,thinhdoan}@vt.edu). This work was supported by the National Science Foundation under ECCS-CAREER Grant No. 2339509.

Prediction accuracy is considered in these methods, however, whether the learned prior and posterior provide trustworthy uncertainty quantification remains an open question.

Contribution statement. In this paper, we consider the problem of Bayesian regression with trustworthy uncertainty quantification. We propose a Bayesian meta learning framework which is cognizant of trustworthy uncertainty quantification without explicit prior assumptions on the model/distribution of the functions. Specifically, we define that the uncertainty quantification is trustworthy if the ground truth can be captured by intervals dependent on the predictive distributions with a pre-specified probability. We then propose Trust-Bayes, a novel optimization framework for Bayesian meta learning with constraints on trustworthy uncertainty quantification using the meta-trained prior distribution and the posterior distribution. We characterize the lower bounds of the probabilities of the ground truth being captured by the specified intervals in terms of the empirical estimates from meta training. We further analyze the sample complexity with respect to the feasible pre-specified probability for trustworthy uncertainty quantification. In summary, our major contributions are threefold:

- We mathematically formulate trustworthy uncertainty quantification for Bayesian regression.
- We propose Trust-Bayes, a novel optimization framework for trustworthy uncertainty quantification.
- We characterize the lower bounds of the probabilities of the ground truth being captured by the specified intervals and analyze the sample complexity with respect to the feasible probability for trustworthy uncertainty quantification.

We conduct Monte Carlo simulation and consider a case study using GPR for verification of trustworthy uncertainty quantification by Trust-Bayes and for comparison against Meta-prior [34]–[36] for its necessary.

Notation. Let $P_x\Big(\mathcal{E}\Big)$ return the probability of event $\mathcal{E}$ with respect to the probability measure of x and $\mathbb{E}_x[\cdot]$ return the expected value with respect to the probability measure of x. Define indicator function $\mathbf{1}[\mathcal{E}] = 1$ if event $\mathcal{E}$ is true and $\mathbf{1}[\mathcal{E}] = 0$ otherwise.

II. Problem formulation

Observation model. Consider a distribution of functions $\mathcal{P}_f$, where each unknown function $f^i \sim \mathcal{P}_f$, $f^i : \mathcal{R}^{n_x} \to \mathbb{R}$, can be observed as

$$y_t^i = f^i(x_t^i), \tag{1}$$

where $x_t^i \in \mathcal{X} \subset \mathcal{R}^{n_x}$ is the tth input to function f^i, n_x is the dimensionality of x_t, and $y_t^i \in \mathbb{R}$ is the corresponding output. In this paper, we consider noiseless observation or the noise is inherent in f^i, which is a typical model for system identification [17] [39] [40]. Here, f^i can be different dynamic models of a system when operating in different environments, such as the dynamics of an autonomous car when operating in different weather conditions.

Regression with uncertainty quantification. Denote the dataset of observations for function f^i as $\mathcal{D}_{tr}^i \triangleq \{(y_t^i, x_t^i)\}_{t=1}^{t_{tr}^i}$, where t_{tr}^i depends on f^i. Denote prior mean function $m : \mathcal{R}^{n_x} \to \mathbb{R}$, prior covariance function $k : \mathcal{R}^{n_x} \times \mathcal{R}^{n_x} \to \mathbb{R}_{>0}$ which is positive semidefinite. Given dataset $\mathcal{D}_{tr}^i$, we denote the corresponding posterior mean function $\mu_{m,k}^i : \mathcal{R}^{n_x} \to \mathbb{R}$ and posterior standard deviation function $\sigma_k^i : \mathcal{R}^{n_x} \to \mathbb{R}$, which are output from some algorithm $\mathcal{ALG}(m, k, \mathcal{D}_{tr}^i)$. The goal of this paper is to identify prior functions m and k which minimize some objective function $J(m, k)$ and meanwhile capture the ground truth $f^i(x)$ by intervals

$$\mathcal{I}_{m,k}(x) \triangleq [m(x) - q\sqrt{k(x,x)}, m(x) + q\sqrt{k(x,x)}]$$
$$\mathcal{I}_{m,k}^i(x) \triangleq [\mu_{m,k}^i(x) - q\sigma_k^i(x), \mu_{m,k}^i(x) + q\sigma_k^i(x)]$$

with a pre-specified probability at least $1 - \delta$, $\delta \in [0, 1]$ for any $x \in \mathcal{X}$. Notice that constants q and δ are fixed a priori, and they do not need to satisfy the relation between reliability factor and level-of-confidence in the settings of confidence interval. Formally, the above specifications for regression with uncertainty quantification is formulated as the optimization problem below

$$\min_{m,k} \ J(m, k) \tag{2a}$$
$$\text{s.t. } P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{m,k}(x)\Big) \geqslant 1 - \delta \tag{2b}$$
$$P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{m,k}^i(x)\Big) \geqslant 1 - \delta \tag{2c}$$
$$\mu_{m,k}^i, \sigma_k^i = \mathcal{ALG}(m, k, \mathcal{D}_{tr}^i). \tag{2d}$$

Constraints (2b) and (2c) aim to ensure trustworthy uncertainty quantification, and one example of the objective function $J(m, k)$ can be negative marginal log likelihood (MLL) [2] evaluated over a dataset as in [34] [35]. Notice that in most cases given an arbitrary function $m(\cdot)$, intervals $\mathcal{I}_{m,k}(\cdot)$ and $\mathcal{I}_{m,k}^i(\cdot)$ can be arbitrarily large, and hence constraints (2b) and (2c) can always be satisfied, by choosing function k such that $k(x, x)$ is sufficiently large if $\mathcal{ALG}$ follows a Bayesian inference framework, e.g., the prior distribution and the likelihood function are Gaussian [5].

We do not assume any additional structure on distribution $\mathcal{P}_f$ or on function f^i, e.g., $f^i \sim \mathcal{GP}(m, k)$, as in [16] [17] [18]. Instead, we assume we have access to a meta dataset $\mathcal{D}^{meta} \triangleq \{\mathcal{D}^i\}_{i=1}^n$, where $\mathcal{D}^i$ contains the observations of $f^i \sim \mathcal{P}_f$ i.i.d. drawn offline, which can be potentially used to estimate and optimize $J(m, k)$ as well as the left hand sides of (2b) and (2c). For testing, we consider functions $f^j \sim \mathcal{P}_f$, which are not necessarily observed in $\mathcal{D}^{meta}$.

Remark II.1. *(Motivation of formulation* (2)*).* Problem (2) is motivated by the problems of safe learning/exploration [17] [14] [34] [24] [41], where the system is required to safely explore and online learn about an unknown environment using the data collected along the system's operation. During the early stage of exploration where there are only a few (or even no) collected data, the control of the system is mainly based on the prior knowledge, i.e., (m, k), of the environment. Therefore, it is crucial that the selected prior is trustworthy such that the ground truth of function f^i can be captured a priori by $\mathcal{I}_{m,k}$, to ensure system safety when the system just starts to explore the environment and no data is collected, and a posteriori by $\mathcal{I}^i_{m,k}$ after collecting a (small) amount of data of f^i. ■

Parameterization. Note that Problem (2) is a functional optimization problem and can be hard to solve in general. To make the problem tractable, we approximate the spaces of m and k using spaces of parameterized functions, e.g., neural networks and finite-dimensional basis functions. Specifically, we consider parameterized prior mean function m_θ, with parameters $\theta \in \mathbb{R}^{n_\theta}$, e.g., the weights of a deep neural network, and parameterized prior mean function k_ϕ, with parameters $\phi \in \mathbb{R}^{n_\phi}$. Then Problem (2) can be rewritten as

$$\min_{\theta,\phi} J(\theta,\phi) \tag{3a}$$

$$\text{s.t. } P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big) \geqslant 1-\delta \tag{3b}$$

$$P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big) \geqslant 1-\delta \tag{3c}$$

$$\mu^i_{\theta,\phi}, \sigma^i_\phi = \mathcal{ALG}(m_\theta, k_\phi, \mathcal{D}^i_{tr}), \tag{3d}$$

with the terms rewritten accordingly. In particular, we require k_ϕ to be scalable, i.e., $k_\phi \triangleq \phi_1\kappa_{\phi_2}$, where $\phi_1 \in \mathbb{R}_{>0}$, $\phi_2 \in \mathbb{R}^{n_\phi-1}$, $\phi=[\phi_1,\phi_2]$, and $\kappa_{\phi_2}: \mathcal{X}\times\mathcal{X}\to[0,1]$ is a covariance function, such that the sizes of the intervals $\mathcal{I}_{\theta,\phi}(x)$ and $\mathcal{I}^i_{m,k}(x)$ can be arbitrarily large by increasing ϕ_1, if $\mathcal{ALG}$ follows some Bayesian inference frameworks such as GPR [2]. Similar to (2), this implies that there always exists ϕ which satisfies (3b) and (3c), and hence Problem (3) is always feasible. One example of covariance function which satisfies the above requirement is the widely-used square-exponential kernel $k(x,x') = \sigma_f^2\exp(-\frac{\|x-x'\|^2}{2\ell}) + \sigma_e^2\delta_{xx'}$.

III. The Trust-Bayes framework

In this section, since distributions $\mathcal{P}_f$ and possibly $\mathcal{P}_x$ are unknown, we introduce the framework Trust-Bayes, to solve problem (3) leveraging the empirical estimates from meta dataset $\mathcal{D}^{meta}$.

Define 0-1 loss functions

$$c^i_1(x) \triangleq \mathbf{1}[f^i(x) \in \mathcal{I}_{\theta,\phi}(x)]$$
$$c^i_2(x) \triangleq \mathbf{1}[f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)].$$

For each $\mathcal{D}^i \in \mathcal{D}^{meta}$, it is split into $\mathcal{D}^i_{tr} \subset \mathcal{D}^i$ for obtaining the posterior functions $(\mu^i_{\theta,\phi}, \sigma^i_k)$ for predictions and an evaluation dataset $\mathcal{D}^i_{eval} \triangleq \mathcal{D}^i \setminus \mathcal{D}^i_{tr}$ for evaluating the performances of the posterior functions and estimating (3b) and (3c). We write $\mathcal{D}^i_{eval} = \{(y^i_t, x^i_t)\}_{t=1}^{t^i_{eval}}$ and $\mathcal{D}^i_{tr} = \{(y^i_t, x^i_t)\}_{t=1}^{t^i_{tr}}$. For any $\gamma \in (0, 0.5]$, define

$$p_1(\gamma) \triangleq \Big(1-2\gamma\Big)\cdot\Big(\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}}\sum_{t=1}^{t^i_{eval}} c^i_1(x^i_t) - \sqrt{\frac{\log(2/\gamma)\sum_{i=1}^n \frac{1}{t^i_{eval}}}{2n^2}} - \sqrt{\frac{\log(2/\gamma)}{2n}}\Big),$$

$$p_2(\gamma) \triangleq \Big(1-2\gamma\Big)\cdot\Big(\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}}\sum_{t=1}^{t^i_{eval}} c^i_2(x^i_t) - \sqrt{\frac{\log(2/\gamma)\sum_{i=1}^n \frac{1}{t^i_{eval}}}{2n^2}} - \sqrt{\frac{\log(2/\gamma)}{2n}}\Big).$$

Then the following theorem characterizes the lower bounds of $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$ and $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$.

Theorem III.1. Suppose $\{f^i\}_{i=1}^n$ are sampled i.i.d. from a latent distribution $\mathcal{P}_f$. Suppose the training dataset $\mathcal{D}^i_{tr}$ for each function f^i is generated through a latent conditional distribution $\mathcal{P}_{\mathcal{D}^i_{tr}|f^i}$, i.e., $\mathcal{D}^i_{tr} \sim \mathcal{P}_{\mathcal{D}^i_{tr}|f^i}$. Suppose for each f^i, $\{x^i_t\}_{t=1}^{t^i_{eval}}$ are sampled i.i.d. from a latent function $\mathcal{P}_x$. Then the following inequalities hold:

$$P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big) \geqslant \max_{\gamma\in(0,0.5]} p_1(\gamma),$$
$$P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big) \geqslant \max_{\gamma\in(0,0.5]} p_2(\gamma).$$

■

The proof of the theorem can be found in [42]. The lower bounds in Theorem III.1 indicates that $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$ and $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$ can be lower bounded by their corresponding empirical estimates of (i.e., $\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}}\sum_{t=1}^{t^i_{eval}} c^i_1(x^i_t)$ and $\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}}\sum_{t=1}^{t^i_{eval}} c^i_2(x^i_t)$) using the evaluation datasets $\mathcal{D}^i_{eval}$, $i=1,\cdots,n$, in the meta training dataset $\mathcal{D}^{meta}$, and the error terms diminish by t^i_{eval} and n, the sizes of $\mathcal{D}^i_{eval}$ and the number of functions in $\mathcal{D}^{meta}$. This provides empirical underestimates of $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$ and $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$ but sufficient verification for constraints (3b) and (3c).

By the lower bounds in the theorem above, we can approximate Problem (3) with

$$\min_{\theta,\phi} J(\theta,\phi) \tag{4a}$$

$$\text{s.t. } \max_{\gamma\in(0,0.5]} p_1(\gamma) \geqslant 1-\delta, \tag{4b}$$

$$\max_{\gamma\in(0,0.5]} p_2(\gamma) \geqslant 1-\delta, \tag{4c}$$

$$\mu^i_{\theta,\phi}, \sigma^i_\phi = \mathcal{ALG}(m_\theta, k_\phi, \mathcal{D}^i_{tr}), \tag{4d}$$

which is our proposed formulation of Trust-Bayes.

Remark III.1. *(Feasibility and sample complexity).* As discussed below Problem (3), the sizes of intervals $\mathcal{I}_{\theta,\phi}(x)$ and $\mathcal{I}^i_{\theta,\phi}(x)$ can be arbitrarily large by choosing ϕ accordingly. This implies that $\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}}\sum_{t=1}^{t^i_{eval}} c^i_j(x_t)$, $j\in\{1,2\}$, can be as large as one. Therefore, by the forms of (4b) and (4c), the feasibility of (4) can be checked by verifying whether

$$\max_{\gamma\in(0,0.5]}(1-2\gamma)\Big(1-\sqrt{\frac{\log(2/\gamma)\sum_{i=1}^n \frac{1}{t^i_{eval}}}{2n^2}} - \sqrt{\frac{\log(2/\gamma)}{2n}}\Big) \geqslant 1-\delta \tag{5}$$

holds for a given δ, which can be done through numerically solving the left hand side using, e.g., gradient ascent. Note that for any $\gamma\in(0,0.5]$, $\sqrt{\frac{\log(2/\gamma)\sum_{i=1}^n \frac{1}{t^i_{eval}}}{2n^2}}$ and $\sqrt{\frac{\log(2/\gamma)}{2n}}$

can be arbitrarily small by increasing sample sizes t^i_{eval} and n. Therefore, inequality (5) provides the sample complexity for the lower bound of feasible δ and implies that the freedom of selecting δ can be increased by increasing t^i_{eval} and n. Note that (4b) and (4c) approximate the probability constraints (3b) and (3c) by underestimating the probabilities with the corresponding empirical estimates. This is reminiscent of the no-free-lunch property between estimation error and the number of data in statistical learning theory [21]. ■

IV. CASE STUDY

In this section, we aim to verify whether the Trust-Bayes formulation in (4) can provide trustworthy uncertainty quantification, i.e., satisfying the constraints in (3), and whether it is necessary. In this case study, we consider GPR [2] as the algorithm $\mathcal{ALG}$. Then the prior predictive distribution for $f^i(x)$ is given by $\mathcal{N}(m_\theta(x), k_\phi(x,x))$ and the posterior predictive distribution is given by $\mathcal{N}(\mu^i_{\theta,\phi}(x), (\sigma^i_\phi)^2(x))$, where

$$\mu^i_{\theta,\phi}(x) = m_\theta(x) + k_\phi(x, X^i)k_\phi^{-1}(X^i, X^i)(Y^i - m_\theta(X^i))$$
$$(\sigma^i_\phi)^2(x) = k_\phi(x,x) + k_\phi(x, X^i)k_\phi^{-1}(X^i, X^i)k_\phi(X^i, x).$$

where X^i aggregates all the inputs x^i_t in $\mathcal{D}^i$, Y^i aggregates all the outputs.

Experiment setup. In this experiment, we let $x \in [0,1]$ and for each $f^i \in \mathcal{P}_f$, $f^i(x) = d^i x^2 + \sum_{m=1}^{10} \alpha^i a^i_m \sin(w^i_m x + \beta^i_m) + (1-\alpha^i) b^i_m \sin(u^i_m x + \beta^i_m)$, where

$$a^i_m \sim 0.5\mathcal{N}(-20, 5) + 0.5\mathcal{N}(10, 2) \quad (6a)$$
$$b^i_m \sim 0.5\mathcal{N}(-1, 0.1) + 0.5\mathcal{N}(1, 0.1) \quad (6b)$$
$$w^i_m \sim 0.5\mathcal{N}(-10, 10) + 0.5\mathcal{N}(10, 10) \quad (6c)$$
$$u^i_m \sim 0.5\mathcal{N}(-100, 10) + 0.5\mathcal{N}(100, 10) \quad (6d)$$
$$\beta^i_m \sim \mathcal{N}(0, 1) \quad (6e)$$
$$d^i \sim 0.5\mathcal{N}(-10, 1) + 0.5\mathcal{N}(10, 1) \quad (6f)$$
$$\alpha^i \sim \mathcal{B}(0.5). \quad (6g)$$

$\mathcal{N}$ denotes normal distribution and $\mathcal{B}$ denotes Bernoulli distribution. For learning using GPR, we consider constant $m_\theta(x) = \theta$ for the prior mean function and kernel $k_\phi(x, x') = \phi_1^2 \exp(-\|x - x'\|^2 \phi_2)$. For trustworthy uncertainty quantification, we select $\delta = 0.1$ and $q_\delta = 1.64$ for intervals $\mathcal{I}_{\theta,\phi}(x)$ and $\mathcal{I}^i_{\theta,\phi}(x)$. That is, we require the predicted intervals with 90% nominal confidence level, according to the Z-score table, to indeed include at least 90% of the true values. Note that according to (6), there are parameters following Gaussian mixtures distributions and Bernoulli distribution, and therefore f^i does not follow a Gaussian process. Furthermore, each f^i has a trend, and therefore $k_\phi(x, x')$ is not a suitable prior covariance function. The purpose of this setup is to demonstrate that Trust-Bayes is able to provide trustworthy uncertainty quantification even under such mis-specification of the prior distribution, which can happen when there is no much prior knowledge on the distribution and structure on the target function f^i.

Training. For the training dataset $\mathcal{D}^{tr}$, we sample 2000 functions f^i, i.e., $n = 2000$, following (6), and for each corresponding training dataset $\mathcal{D}^i$ we uniformly sample over interval $[0,1]$ for inputs $t^i_{tr} = 20$ for obtaining the posterior and $t^i_{eval} = 100$ for evaluating the posterior for (4a) to (4c). Note that with the above specification of δ, n, and t^i_{eval}, (5) holds and can be verified by plugging in $\gamma = 0.001$. For the objective function $J(\theta, \phi)$, we consider negative MLL over the whole meta dataset $\mathcal{D}^{tr}$ as in [34]–[36].

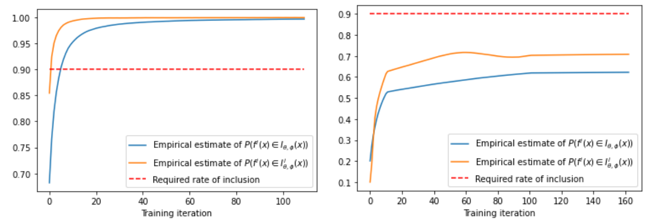
Testing. Notice that the probabilities in (3b) and (3c) can be challenging to exactly obtain. As an estimation, the trained hyperparameters (θ, ϕ) are tested against a dataset composed of $10,000$ functions f^i randomly sampled over (6) with each f^i has testing $10,000$ inputs x uniformly sampled over $[0,1]$. The corresponding posterior distribution is obtained after observing each function over 20 inputs without further tuning the hyperparameters.

Comparison. Our results are compared against the Meta-prior method used in [34]–[36], where the hyperparameters in the prior distributions are meta-trained using negative MLL. Note that the Bayesian regression approach used in [34] [35] is a special case of GPR [2]. The difference between Trust-Bayes and Meta-prior in this case study is the addition of (4b) and (4c).

Results. Figure 1 shows the convergence of $\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}} \sum_{t=1}^{t^i_{eval}} c^i_1(x^i_t)$ and $\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}} \sum_{t=1}^{t^i_{eval}} c^i_2(x^i_t)$, the empirical estimates of $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$ and $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$, respectively, by the evaluation dataset $\{\mathcal{D}^i_{eval}\}_{i=1}^n$ in the meta dataset $\mathcal{D}^{meta}$. From Figure 1, we can see that the training of Meta-prior converges when the empirical estimates are still at a level much lower than the required rate of inclusion $1 - \delta = 0.9$. In contrast, the training of Trust-Bayes converges much faster and beyond the required rate of inclusion.

Table I provides a comparison between the (θ, ϕ) trained till convergence by Trust-Bayes and that by Meta-prior over the estimated $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$, $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$, their empirical estimates by $\mathcal{D}^i_{eval}$ as well as the mean-squared error (MSE) for the posterior predictions over the testing dataset. From Table I, we can see that Trust-Bayes performs better than Meta-prior over all the metrics. Specifically, $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\Big)$ and $P_{f^i,x}\Big(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\Big)$ are higher than the required inclusion rate (i.e, 0.9) when (θ, ϕ) are trained by Trust-Bayes, which verifies that Trust-Bayes can provide trustworthy uncertainty quantification. In contrast, when (θ, ϕ) are trained by Meta-prior, the rates of inclusion are much lower than the required inclusion rate although the intervals are constructed with 90% nominal level of confidence. This comparison with Meta-prior shows the necessary of using Trust-Bayes to provide trustworthy uncertainty quantification when it is uncertain whether the prior is properly specified or not.

Figure 2 provides a visual comparison over the inclusions of the true values of $f^i(x)$ by the prior 90% confidence intervals $\mathcal{I}_{\theta,\phi}(x)$ trained using Trust-Bayes and Meta-prior, respectively, and Figure 3 provides a comparison over the



(a) Trust-Bayes (b) Meta-prior

Fig. 1: Comparison over empirical estimates of $P_{f^i,x}\left(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\right)$ and $P_{f^i,x}\left(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\right)$ by the evaluation dataset $\{\mathcal{D}^i_{eval}\}_{i=1}^n$ between Trust-Bayes and Meta-prior during meta-training

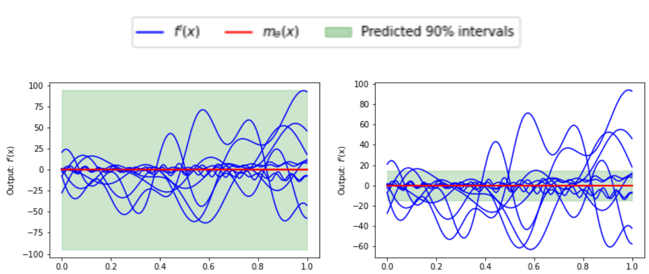
	Trust-Bayes	Meta-prior
$\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}} \sum_{t=1}^{t^i_{eval}} c^i_1(x^i_t)$	0.997	0.623
$\frac{1}{n}\sum_{i=1}^n \frac{1}{t^i_{eval}} \sum_{t=1}^{t^i_{eval}} c^i_2(x^i_t)$	0.999	0.706
$P_{f^i,x}\left(f^i(x) \in \mathcal{I}_{\theta,\phi}(x)\right)$	0.996	0.630
$P_{f^i,x}\left(f^i(x) \in \mathcal{I}^i_{\theta,\phi}(x)\right)$	0.999	0.708
MSE	91.42	125.15

TABLE I: Comparisons over empirical and expected prior/posterior inclusions between Trust-Bayes and Meta-prior

inclusions by the posterior 90% confidence intervals $\mathcal{I}^i_{\theta,\phi}(x)$. Ten functions are sampled from (6) to provide a visualization of the distribution of the functions. From these two figures, we can see that the 90% intervals generated by Trust-Bayes are larger and are able to capture most of the values of the functions including those with larger variation, whereas Meta-prior generates smaller intervals and fails to capture the values of the functions with larger variation.

V. Conclusion

We consider trustworthy uncertainty quantification in Bayesian regression problems. We formulate trustworthy uncertainty quantification as constraints on capturing the ground truths of the function by intervals depending on the prior and posterior distributions with a pre-specified probability. We propose, Trust-Bayes, a Bayesian meta learning framework which is cognizant of trustworthy uncertainty quantification without explicit assumptions on the model/distribution of the functions. We characterize the lower bounds of the probabilities of the ground truth being captured by the specified intervals in terms of the empirical estimates and analyze the sample complexity with respect to the feasible pre-specified probability for trustworthy uncertainty quantification. Monte Carlo simulation is conducted for evaluation and comparison through a case study using GPR, which verifies the proposed framework and demonstrates the necessary of Trust-Bayes for trustworthy uncertainty quantification when the prior is not necessarily correctly specified.

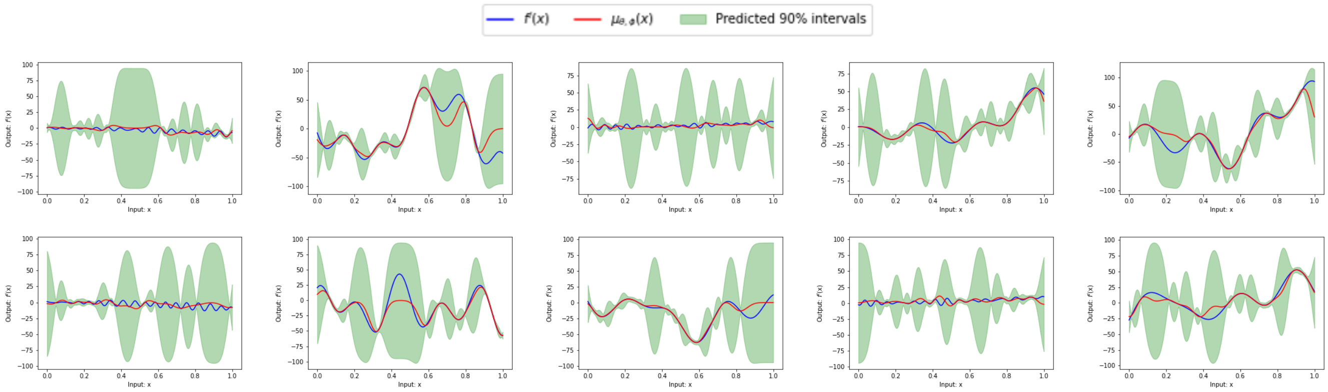


(a) Trust-Bayes (b) Meta-prior

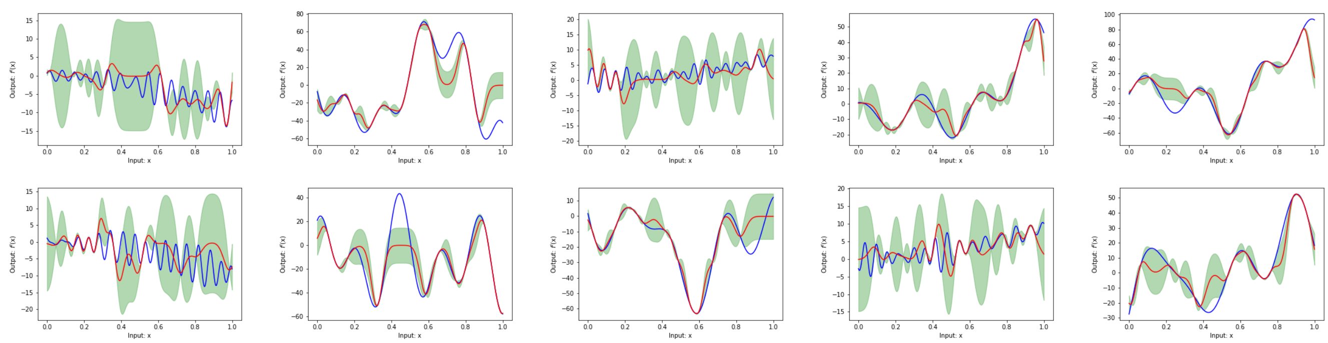
Fig. 2: Ten random functions drawn from (6) and comparison between prior predictions by Trust-Bayes and Meta-prior

References

[1] T. M. Mitchell, "Machine learning," 1997.
[2] C. K. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
[3] G. Welch, G. Bishop *et al.*, "An introduction to the kalman filter," 1995.
[4] J. Lampinen and A. Vehtari, "Bayesian approach for neural networks—review and case studies," *Neural networks*, vol. 14, no. 3, pp. 257–274, 2001.
[5] C. M. Bishop, *Pattern recognition*. Springer, 2006.
[6] T. Choi and M. J. Schervish, "On posterior consistency in nonparametric regression problems," *Journal of Multivariate Analysis*, vol. 98, no. 10, pp. 1969–1987, Jan. 2007.
[7] G. E. Box and G. C. Tiao, *Bayesian inference in statistical analysis*. John Wiley & Sons, 2011.
[8] D. A. McAllester, "Some PAC-Bayesian theorems," *Machine Learning*, vol. 37, no. 3, pp. 355–363, 1999.
[9] R. Garnett, *Bayesian optimization*. Cambridge University Press, 2023.
[10] A. Lederer, Z. Yang, J. Jiao, and S. Hirche, "Cooperative control of uncertain multi-agent systems via distributed gaussian processes," *IEEE Transactions on Automatic Control*, 2022.
[11] M. Liu, G. Chowdhary, B. C. Da Silva, S.-Y. Liu, and J. P. How, "Gaussian processes for learning and control: A tutorial with examples," *IEEE Control Systems Magazine*, vol. 38, no. 5, pp. 53–86, 2018.
[12] N. Virani, D. K. Jha, Z. Yuan, I. Shekhawat, and A. Ray, "Imitation of demonstrations using bayesian filtering with nonparametric data-driven models," *Journal of Dynamic Systems, Measurement, and Control*, vol. 140, no. 3, 2018.
[13] M. Mukadam, X. Yan, and B. Boots, "Gaussian process motion planning," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 9–15.
[14] Z. Yuan and M. Zhu, "dSLAP: Distributed safe learning and planning for multi-robot systems," in *Proc. IEEE Conf. Decision and Control (CDC)*, 2022, pp. 5864–5869.
[15] A. Chiuso and G. Pillonetto, "A bayesian approach to sparse dynamic network identification," *Automatica*, vol. 48, no. 8, pp. 1553–1565, 2012.
[16] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for Gaussian process optimization in the bandit setting," *IEEE Trans. Information Theory*, vol. 58, no. 5, pp. 3250–3265, Jan. 2012.
[17] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems," *IEEE Trans. Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2019.
[18] Z. Yuan and M. Zhu, "Lightweight distributed gaussian process regression for online machine learning," *IEEE Transactions on Automatic Control*, 2024. To appear.
[19] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing, "Deep kernel learning," in *Artificial intelligence and statistics*. PMLR, 2016, pp. 370–378.
[20] Y. Shen, T. Chen, and G. B. Giannakis, "Random feature-based online multi-kernel learning in environments with unknown dynamics," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 773–808, 2019.



(a) Posterior predictions over ten sample functions by Trust-Bayes



(b) Posterior predictions over ten sample functions by Meta-prior

Fig. 3: Comparison between posterior predictions by Trust-Bayes and Meta-prior over ten random functions from (6)

[21] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer science & business media, 2013.

[22] Z. Zhou, O. S. Oguz, M. Leibold, and M. Buss, "A general framework to increase safety of learning algorithms for dynamical systems based on region of attraction estimation," *IEEE Trans. Robotics*, 2020.

[23] J. Umlauft, T. Beckers, A. Capone, A. Lederer, and S. Hirche, "Smart forgetting for safe online learning with gaussian processes," in *Learning for dynamics and control*, 2020, pp. 160–169.

[24] F. Berkenkamp, A. Krause, and A. P. Schoellig, "Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics," *Machine Learning*, vol. 112, no. 10, pp. 3713–3747, 2023.

[25] Y. Sui, V. Zhuang, J. Burdick, and Y. Yue, "Stagewise safe bayesian optimization with gaussian processes," in *Proc. Int. Conf. Machine Learning (ICML)*, 2018, pp. 4781–4789.

[26] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.

[27] A. Rajeswaran, C. Finn, S. M. Kakade, and S. Levine, "Meta-learning with implicit gradients," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[28] M. Khodak, M.-F. F. Balcan, and A. S. Talwalkar, "Adaptive gradient-based meta-learning methods," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[29] J. Lee, J. Tack, N. Lee, and J. Shin, "Meta-learning sparse implicit neural representations," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 769–11 780, 2021.

[30] S. Xu and M. Zhu, "Online constrained meta-learning: Provable guarantees for generalization," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[31] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Generalization of model-agnostic meta-learning algorithms: Recurring and unseen tasks," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 5469–5480, 2021.

[32] J. Yoon, T. Kim, O. Dia, S. Kim, Y. Bengio, and S. Ahn, "Bayesian model-agnostic meta-learning," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[33] X. Zhang, D. Meng, H. Gouk, and T. M. Hospedales, "Shallow bayesian meta learning for real-world few-shot recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 651–660.

[34] T. Lew, A. Sharma, J. Harrison, A. Bylard, and M. Pavone, "Safe active dynamics learning and control: A sequential exploration–exploitation framework," *IEEE Trans. Robotics*, vol. 38, no. 5, pp. 2888–2907, 2022.

[35] J. Harrison, A. Sharma, and M. Pavone, "Meta-learning priors for efficient online bayesian regression," in *Algorithmic Foundations of Robotics XIII: Proceedings of the 13th Workshop on the Algorithmic Foundations of Robotics 13*, 2020, pp. 318–337.

[36] J. Rothfuss, V. Fortuin, M. Josifoski, and A. Krause, "Pacoh: Bayes-optimal meta-learning with pac-guarantees," in *Proc. Int. Conf. Machine Learning (ICML)*, 2021, pp. 9116–9126.

[37] M. Patacchiola, J. Turner, E. J. Crowley, M. O'Boyle, and A. J. Storkey, "Bayesian meta-learning for the few-shot setting via deep kernels," *Advances in Neural Information Processing Systems*, vol. 33, pp. 16 108–16 118, 2020.

[38] S. Nabi, H. Nassif, J. Hong, H. Mamani, and G. Imbens, "Bayesian meta-prior learning using empirical bayes," *Management Science*, vol. 68, no. 3, pp. 1737–1755, 2022.

[39] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the national academy of sciences*, vol. 113, no. 15, pp. 3932–3937, 2016.

[40] V. Dhiman, M. J. Khojasteh, M. Franceschetti, and N. Atanasov, "Control barriers in bayesian learning of system dynamics," *IEEE Trans. Automatic Control*, 2021.

[41] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," in *Proc. Int. Conf. Robotics and Automation (ICRA)*, 2018, pp. 2460–2465.

[42] Z. Yuan and T. T. Doan, "Bayesian meta learning for trustworthy uncertainty quantification," *arXiv preprint arXiv:2407.19287*, 2024.